



## Calhoun: The NPS Institutional Archive

---

Theses and Dissertations

Thesis and Dissertation Collection

---

2016-06

# A new approach to site demand-based level inventory optimization

Ersoz, Tacettin

Monterey, California: Naval Postgraduate School

---

<http://hdl.handle.net/10945/49452>



Calhoun is a project of the Dudley Knox Library at NPS, furthering the precepts and goals of open government and government transparency. All information contained herein has been approved for release by the NPS Public Affairs Officer.

**Dudley Knox Library / Naval Postgraduate School**  
**411 Dyer Road / 1 University Circle**  
**Monterey, California USA 93943**

<http://www.nps.edu/library>



# **NAVAL POSTGRADUATE SCHOOL**

**MONTEREY, CALIFORNIA**

## **THESIS**

**A NEW APPROACH TO SITE DEMAND-BASED LEVEL  
INVENTORY OPTIMIZATION**

by

Tacettin Ersoz

June 2016

Thesis Advisor:  
Second Reader:

Javier Salmeron  
Emily Craparo

**Approved for public release; distribution is unlimited**

THIS PAGE INTENTIONALLY LEFT BLANK

<b>REPORT DOCUMENTATION PAGE</b>			<i>Form Approved OMB No. 0704-0188</i>	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.				
<b>1. AGENCY USE ONLY</b> (Leave blank)	<b>2. REPORT DATE</b> June 2016	<b>3. REPORT TYPE AND DATES COVERED</b> Master's thesis		
<b>4. TITLE AND SUBTITLE</b> A NEW APPROACH TO SITE DEMAND-BASED LEVEL INVENTORY OPTIMIZATION			<b>5. FUNDING NUMBERS</b>	
<b>6. AUTHOR(S)</b> Tacettin Ersoz				
<b>7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)</b> Naval Postgraduate School Monterey, CA 93943-5000			<b>8. PERFORMING ORGANIZATION REPORT NUMBER</b>	
<b>9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)</b> N/A			<b>10. SPONSORING / MONITORING AGENCY REPORT NUMBER</b>	
<b>11. SUPPLEMENTARY NOTES</b> The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol number ____N/A____.				
<b>12a. DISTRIBUTION / AVAILABILITY STATEMENT</b> Approved for public release; distribution is unlimited			<b>12b. DISTRIBUTION CODE</b>	
<b>13. ABSTRACT (maximum 200 words)</b>  <p>Naval Supply Systems Command (NAVSUP) supports Navy, Marine Corps, Joint and Allied Forces with their inventory of more than 430,000 items worth \$21 billion using several distribution sites. Choosing the optimum order-point and order-quantity for each item is important to meet the stochastic demand while satisfying multiple restrictions such as budget and maximum number of orders. The Site Demand-Based Level Inventory Optimization Model (SIOM) is a mixed-integer, linear program developed at the Naval Postgraduate School to provide NAVSUP planners with guidance on this complex problem. Ongoing tests have been successful, but SIOM's computational run times are long. This thesis introduces a new, faster reformulation (SIOMsQ) that approximates the solution of the same problem by reducing the possible candidate sets of order-points and order-quantities for each item. We find that the solutions suggested by SIOMsQ are better than or very close to those of SIOM in test cases provided by NAVSUP, with substantially shorter computational times. Therefore, we recommend using SIOMsQ versus SIOM.</p>				
<b>14. SUBJECT TERMS</b> inventory management, inventory optimization, order points, order quantities, Naval Supply Systems Command, demand based item, mixed-integer linear problem			<b>15. NUMBER OF PAGES</b> 63	
			<b>16. PRICE CODE</b>	
<b>17. SECURITY CLASSIFICATION OF REPORT</b> Unclassified	<b>18. SECURITY CLASSIFICATION OF THIS PAGE</b> Unclassified	<b>19. SECURITY CLASSIFICATION OF ABSTRACT</b> Unclassified	<b>20. LIMITATION OF ABSTRACT</b> UU	

THIS PAGE INTENTIONALLY LEFT BLANK

**Approved for public release; distribution is unlimited**

**A NEW APPROACH TO SITE DEMAND-BASED LEVEL INVENTORY  
OPTIMIZATION**

Tacettin Ersoz  
Lieutenant Junior Grade, Turkish Navy  
B.S., Turkish Naval Academy, 2009

Submitted in partial fulfillment of the  
requirements for the degree of

**MASTER OF SCIENCE IN OPERATIONS RESEARCH**

from the

**NAVAL POSTGRADUATE SCHOOL  
June 2016**

Approved by: Javier Salmeron  
Thesis Advisor

Emily Craparo  
Second Reader

Patricia Jacobs  
Chair, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

## **ABSTRACT**

Naval Supply Systems Command (NAVSUP) supports Navy, Marine Corps, Joint and Allied Forces with their inventory of more than 430,000 items worth \$21 billion using several distribution sites. Choosing the optimum order-point and order-quantity for each item is important to meet the stochastic demand while satisfying multiple restrictions such as budget and maximum number of orders. The Site Demand-Based Level Inventory Optimization Model (SIOM) is a mixed-integer, linear program developed at the Naval Postgraduate School to provide NAVSUP planners with guidance on this complex problem. Ongoing tests have been successful, but SIOM's computational run times are long. This thesis introduces a new, faster reformulation (SIOMsQ) that approximates the solution of the same problem by reducing the possible candidate sets of order-points and order-quantities for each item. We find that the solutions suggested by SIOMsQ are better than or very close to those of SIOM in test cases provided by NAVSUP, with substantially shorter computational times. Therefore, we recommend using SIOMsQ versus SIOM.



THIS PAGE INTENTIONALLY LEFT BLANK

## TABLE OF CONTENTS

<b>I.</b>	<b>INTRODUCTION.....</b>	<b>1</b>
<b>A.</b>	<b>BACKGROUND .....</b>	<b>1</b>
<b>B.</b>	<b>REVIEW OF INVENTORY MODELS.....</b>	<b>2</b>
<b>C.</b>	<b>PROBLEM STATEMENT .....</b>	<b>4</b>
<b>D.</b>	<b>RESEARCH OBJECTIVES AND SCOPE.....</b>	<b>6</b>
<b>II.</b>	<b>METHODOLOGY .....</b>	<b>7</b>
<b>A.</b>	<b>SIOM CHARACTERISTICS.....</b>	<b>7</b>
<b>B.</b>	<b>SIOMSQ DEVELOPMENT .....</b>	<b>8</b>
1.	Definitions.....	8
2.	Generation of Parameters $s$ and $Q$ .....	8
<b>C.</b>	<b>MATHEMATICAL MODEL .....</b>	<b>9</b>
1.	Indices and Index Sets .....	9
2.	Input Data and Parameters [Units].....	10
3.	Decision Variables.....	11
4.	Derived Data.....	11
5.	SIOMsQ Formulation.....	14
6.	Two-Step Approach.....	15
<b>D.</b>	<b>COMPUTATIONAL EXPERIENCE.....</b>	<b>15</b>
<b>III.</b>	<b>COMPUTATIONAL ANALYSIS.....</b>	<b>17</b>
<b>A.</b>	<b>RUNNING OPTIONS FOR SIOM AND SIOMSQ.....</b>	<b>17</b>
<b>B.</b>	<b>RESULTS .....</b>	<b>18</b>
1.	Yokosuka (NMC) .....	18
2.	Key West (BP28) .....	19
3.	Kings Bay (NMC).....	20
4.	San Diego (NMC) .....	21
5.	Norfolk (NMC) .....	22
6.	Jacksonville (BP28).....	23
7.	Kings Bay (BP28) .....	24
8.	Norfolk (BP28) .....	25
9.	Bangor (BP28) .....	26
10.	Yokosuka (BP28).....	27
<b>C.</b>	<b>SUMMARY OF RESULTS .....</b>	<b>29</b>
<b>IV.</b>	<b>CONCLUSIONS AND FUTURE RESEARCH.....</b>	<b>31</b>
<b>A.</b>	<b>CONCLUSIONS .....</b>	<b>31</b>

B.	FUTURE RESEARCH.....	31
APPENDIX.	SIOM FORMULATION .....	33
A.	INDICES AND INDEX SETS .....	33
B.	INPUT DATA AND PARAMETERS .....	33
C.	DECISION VARIABLES .....	36
D.	FORMULATION.....	37
LIST OF REFERENCES.....		39
INITIAL DISTRIBUTION LIST .....		41

## LIST OF FIGURES

Figure 1.	$(s, Q)$ Inventory System. Source: Silver et al. (1998).....	3
-----------	--	---

THIS PAGE INTENTIONALLY LEFT BLANK

## LIST OF TABLES

Table 1.	Yokosuka (NMC) SIOM Results.....	18
Table 2.	Yokosuka (NMC) SIOMsQ Results .....	19
Table 3.	Key West (BP28) SIOM Results .....	19
Table 4.	Key West (BP28) SIOMsQ Results.....	20
Table 5.	Kings Bay (NMC) SIOM Results.....	20
Table 6.	Kings Bay (NMC) SIOMsQ Results.....	21
Table 7.	San Diego (NMC) SIOM Results .....	21
Table 8.	San Diego (NMC) SIOMsQ Results.....	22
Table 9.	Norfolk (NMC) SIOM Results .....	22
Table 10.	Norfolk (NMC) SIOMsQ Results.....	23
Table 11.	Jacksonville (BP28) SIOM Results .....	23
Table 12.	Jacksonville (BP28) SIOMsQ Results.....	24
Table 13.	Kings Bay (BP28) SIOM Results .....	24
Table 14.	Kings Bay (BP28) SIOMsQ Results.....	25
Table 15.	Norfolk (BP28) SIOM Results.....	25
Table 16.	Norfolk (BP28) SIOMsQ Results.....	26
Table 17.	Bangor (BP28) SIOM Results .....	27
Table 18.	Bangor (BP28) SIOMsQ Results.....	27
Table 19.	Yokosuka (BP28) SIOM Results.....	28
Table 20.	Yokosuka (BP28) SIOMsQ Results.....	28
Table 21.	Results Summary of Test Cases.....	30

THIS PAGE INTENTIONALLY LEFT BLANK

## LIST OF ACRONYMS AND ABBREVIATIONS

BP28	Navy Working Capital Fund Budget Program 28
GAMS	General Algebraic Modeling System
GHA	Grouping Heuristic Algorithm
MINLP	Mixed-Integer Non-linear Problem
MSL	Maximum Stock Level
NAVSUP	Naval Supply Systems Command
NAVSUP WSS	Naval Supply Systems Command Weapon Systems Support
NIIN	National Item Identification Number
NMC	Navy-Managed Consumables
OPNAVINST	Chief of Naval Operations Instruction
SIOM	Site Demand-Based Level Inventory Optimization Model
SIOMsQ	Site Demand-Based Level Inventory Optimization Model with pre-generated order-points, $s$ , and order-quantities, $Q$
SPO	Service Planning and Optimization
WIOM	Wholesale Inventory Optimization Model



THIS PAGE INTENTIONALLY LEFT BLANK

## EXECUTIVE SUMMARY

The mission of the Naval Supply Systems Command (NAVSUP) is to provide the Navy and Joint Warfighter with global logistics and quality-of-life support (U.S. Naval Academy 2016). NAVSUP Weapon Systems Support (NAVSUP WSS) is responsible for a large fraction of NAVSUP's National Item Identification Numbers (NIINs) with more than 430,000 items (NAVSUP 2016). As its primary mission, NAVSUP WSS has to provide all the Navy and Marine Corps units with the items they need with minimal or no delay.

In order to maximize the operational readiness of hundreds of units, NAVSUP WSS needs to keep adequate wholesale and retail inventory levels for all items subject to multiple constraints. As a measure of effectiveness, “fill rate” is used to specify the expected fraction of orders for which a replacement is available (i.e., on hand when demand occurs). Achieving a good fill rate requires choosing an order-point,  $s$ , and an order-quantity,  $Q$ , for each item.

In order to find optimum order-points and order-quantities, Salmeron and Craparo (2016) have developed Site Demand-Based Level Inventory Optimization Model (SIOM) to guide stock level decisions for NAVSUP. Because of the large scale and non-linear aspect of the problem, SIOM pre-generates candidate order-quantities for every NIIN, and then optimizes order-points and order-quantities (restricted to pre-generated ones). At the time of this research, SIOM is undergoing testing for ten test cases at NAVSUP WSS. Preliminary results are promising, albeit computational times are long. This thesis develops SIOMsQ, an efficient reformulation of SIOM which improves both solution quality and computational time.

SIOMsQ approximates the solution of SIOM by reducing the possible candidate sets of order-points and order-quantities for each item. That is, we proceed as in SIOM but limit the values of, not only  $Q$ , but also  $s$ , to a list of pre-generated candidate  $(s, Q)$  pairs. For example, SIOMsQ can generate up to 10 candidate order-quantities for every NIIN, and then 20 candidate order-points for each order-quantity and NIIN. First, we set

upper and lower limits of both  $s$  and  $Q$  using the data provided by NAVSUP (shelf life, maximum stock level, average monthly demand, etc.). Then we develop a strategy to select the candidate order-points and order-quantities between their respective limits, such that the differences between consecutive candidates are approximately the same.

SIOMsQ is formulated as a pure integer, linear program that chooses exactly one candidate for each item, so that the overall choice is feasible for all items. The objective of SIOMsQ is to minimize the total expected penalty by choosing the optimum (already generated)  $(s, Q)$  pair for each NIIN while satisfying the constraints such as budget, maximum number of orders per month and shelf life among others. Since, for each generated  $(s, Q)$  pair, we can pre-calculate the associated fill rate, cost, penalties, and other data that would apply should such pair become the decision adopted for the item, the resulting model does not need the complex constructs of SIOM. The possibly large number of binary decision variables due to candidate pairs is compensated by a reduction in model complexity, as proven by our computational results.

SIOMsQ is run in General Algebraic Modeling System (GAMS) using CPLEX solver (GAMS 2016). We run SIOMsQ with various candidate  $(s, Q)$  pair sizes which range from 7 to 20 candidate  $Q$ -values, and similarly for candidate  $s$ -values for each  $Q$ .

For all of the ten test cases (from real-world problems provided by NAVSUP), the solutions SIOMsQ achieves are better than or very close to those of SIOM, but the running times of SIOMsQ are substantially shorter. The improvements are most compelling in large-scale problems, where SIOM has notable solving difficulties.

As a conclusion of this research, we recommend that running SIOMsQ using either 7  $Q$ -values and 7  $s$ -values for each item (if very fast runs with acceptable solution quality are required), or 10  $Q$ -values and 20  $s$ -values for each item (if solution quality is very important, still with acceptable running times).

## References

GAMS (2016) CPLEX 12. Accessed April 28, 2016,  
<http://www.gams.com/help/index.jsp?topic=%2Fgams.doc%2Fsolvers%2Findex.html>

Naval Supply Systems Command (2016) Navy supply chain management. Accessed April 17, 2016, <https://www.navsup.navy.mil/navsup/capabilities/nscm>

Salmeron J, Craparo E (2016) Site Demand-Based Level Inventory Optimization Model, Release 1.2.3. Project Deliverable, Naval Postgraduate School.

United States Naval Academy (2016) Commanders Guidance. Accessed May 7, 2016, [http://www.usna.edu/Training/\\_files/documents/References/1C%20MQS%20References/NAVSUP%202014%20Commanders%20Guidance.pdf](http://www.usna.edu/Training/_files/documents/References/1C%20MQS%20References/NAVSUP%202014%20Commanders%20Guidance.pdf)

THIS PAGE INTENTIONALLY LEFT BLANK

## **ACKNOWLEDGMENTS**

I would like to extend my warmest appreciation and thanks to my thesis advisor, Professor Javier Salmeron, and second reader, Professor Emily Craparo. I would not have finished this research without their guidance, dedicated support, and patience.

I wish to express my deepest gratitude to the operations research faculty at the Naval Postgraduate School. They have provided me with all the tools to be a skilled military operations research analyst.

Lastly, I would like to thank the Turkish Naval Forces for giving me the opportunity to pursue my further education at such a reputable university as the Naval Postgraduate School.

THIS PAGE INTENTIONALLY LEFT BLANK

# **I. INTRODUCTION**

## **A. BACKGROUND**

The mission of the Naval Supply Systems Command (NAVSUP) is to provide the Navy and Joint Warfighter with global logistics and quality-of-life support (U.S. Naval Academy 2016). Involving about 345 military personnel and more than 24,000 civilian employees, NAVSUP holds inventory for approximately \$21 billion and spends a yearly budget of \$3.5 billion (NAVSUP 2016a, b).

The mission of NAVSUP Weapon Systems Support (NAVSUP WSS) is to deliver weapon systems supply support to the Navy, Marine Corps, Joint and Allied Forces (NAVSUP 2016c). NAVSUP WSS is responsible for a large fraction of NAVSUP's National Item Identification Numbers (NIINs) with more than 430,000 items (NAVSUP 2016a).

As its primary mission, NAVSUP WSS has to provide all the Navy and Marine Corps units with the items they need with minimal or no delay.

In order to maximize the operational readiness of hundreds of units, NAVSUP WSS needs to keep adequate wholesale and retail inventory levels for all items subject to multiple constraints. A key complication in the analysis is the fact that demands for the NIINs are stochastic.

As measure of effectiveness, “fill rate” is defined as the expected fraction of orders for which a replacement is available (i.e., on hand when demand occurs). A detailed formulation is included in Section II.C.

Due to limitations such as inventory budget, number of orders per month, storage space and shelf life, it is not possible to achieve a perfect fill rate of 100% for all NIINs. Instead, NAVSUP WSS specifies minimum required fill rate levels for all items, based for example on the importance of the item to readiness.

This thesis develops a mathematical optimization model to guide inventory decisions for NAVSUP.



## B. REVIEW OF INVENTORY MODELS

Silver et al. (1998) point out the importance of inventory management and production planning and scheduling of several industries and organizations in economic, medical and military aspects. When modeling the inventory management system of individual items with probabilistic demand, Silver et al. introduce two techniques: continuous review and periodic review.

In continuous review, the state of the inventory is assumed to be known at all times; for example, an order can be placed at any point in time. In periodic review, the inventory is verified (and related decisions are made) at discrete points in time (e.g., on a given weekday). For this study, we are solely concerned with continuous review models.

Silver et al. (1998) describes two types of continuous review inventory management systems: order-point, order-quantity ( $s, Q$ ) system and order-point, order-up-to-level ( $s, S$ ) system. (Note: order-point and order-quantity are sometimes referred to as reorder point and order size, respectively. We use the same terminology as in Silver et al. (1998) in the rest of this thesis.)

A backorder is a demand that cannot be filled at the time it occurs (i.e., an outstanding order for which the customer will need to wait). Net stock is defined as the number of items on hand, whereas inventory position includes the number of items ordered and backordered in addition to net stock. Inventory position at any given point in time can be formulated as follows:

$$\text{Inventory Position} = \text{Net Stock} + \text{Orders} - \text{Backorders}$$

In an ( $s, S$ ) system, when the inventory position drops below  $s$ , an order is placed such that the inventory position reaches the order-up-to level,  $S$ . Different amounts of items can be ordered at different times depending on how far the inventory position is away from the order-up-to-level. On the other hand, in an ( $s, Q$ ) system, when the inventory position drops below  $s$ , a pre-determined order amount  $Q$  is placed.

It should be kept in mind that an order is placed when the inventory position drops below order-point,  $s$ . If orders were placed according to the net stock, there could

be unnecessary orders that stem from not waiting the lead time of the previous order (Silver et al. 1998).

In Figure 1, inventory position is represented by dashed line and net stock or both the inventory position and net stock are shown by the solid line. There are demands of different amounts at several points in time. When the inventory drops below  $s$  at time  $A$ , an order is placed and the inventory position increases by  $Q$ .  $L$  represents the order lead time, which is assumed to be deterministic.

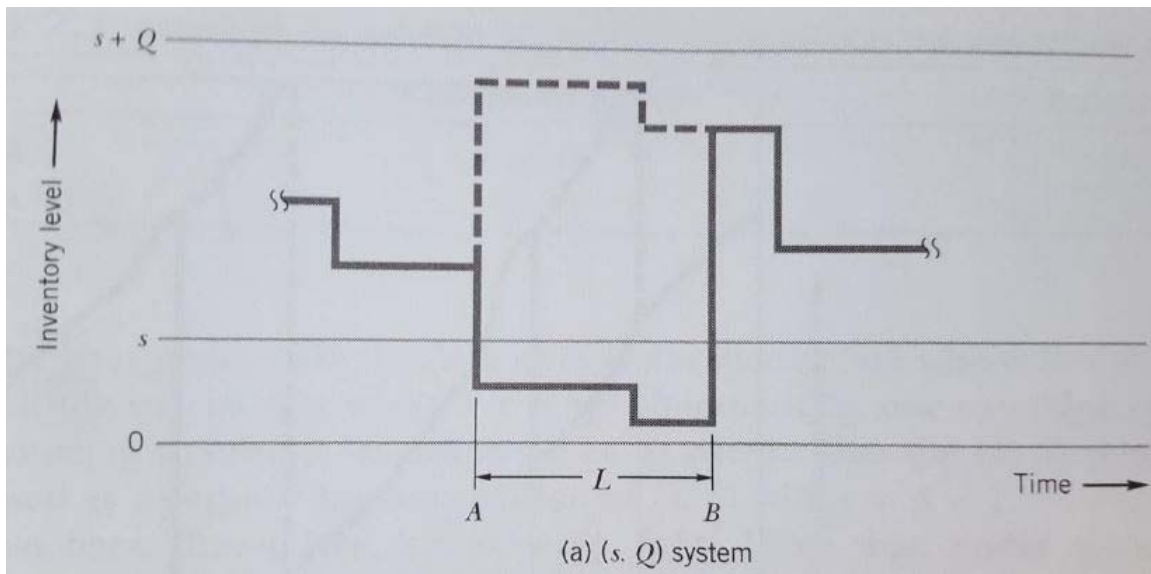


Figure 1.  $(s, Q)$  Inventory System. Source: Silver et al. (1998).

For a certain item, choosing a large order-point provides a good fill rate, but this increases the amount of money tied up in inventory. Also, some items might exhaust their shelf life before they are demanded. On the other hand, if a small order-point is selected, it could cause backorders, thus reducing the fill rate.

Similar to the order-points, order-quantities have an important effect on fill rates. If a small order-quantity is selected for an item, backorders could arise depending upon the lead time (the time elapsed between an order placement and its arrival) of the item. Also, a small order-quantity requires orders to be placed more frequently, incurring

additional costs and preventing potential savings such as discounts from seller. On the other hand, a large order-quantity could cause some items to perish before demanded.

Determining the order-points and order-quantities for thousands of items in order to achieve target fill rates is a complicated task because all the NIINs share a budget limit and an order limit (the number of order documents that can be placed in a given time period).

For the problem at hand, an  $(s, S)$  model is computationally intractable; therefore we develop an  $(s, Q)$  model and approximate the optimal  $(s, S)$  solution by setting  $S = s + Q$ .

Salmeron and Craparo (2015) introduce the Wholesale Inventory Optimization Model (WIOM). Given the order-quantities, WIOM optimizes the order-points of all items simultaneously by minimizing the expected weighted deviation from the target fill rates of each item.

Roth (2016) compares the performances of three different inventory management models using a simulation: simple calculation of fill rates, SPO and WIOM. His results show that the order-points suggested by WIOM provide higher fill rates than the two other models.

The key idea this thesis exploits is the enumeration of possible values of decision variables to enable pre-processing of information with which to build an approximate optimization model. Enumeration is a common technique in optimization. For example, Kolodziej et al. (2013) use this method in optimization of the multi-period blend scheduling problem. In order to overcome the difficulties of the non-convex MINLP problem, they enumerate to approximate the optimum blending fractions of various products.

## **C. PROBLEM STATEMENT**

OPNAVINST 4441.12D defines demand-based items as "items that have a relatively high issue rate" (Office of the Chief of Naval Operations 2012).

NAVSUP WSS is now using Morris Cohen Associates' 'Service Planning and Optimization (SPO)' tool for planning order-points and order-quantities for demand-based items. However, they do not document the mathematical model or algorithms used to determine such values (Roth 2016).

In order to find optimum order-points and order-quantities, Salmeron and Craparo (2016) have developed Site Demand-Based Level Inventory Optimization Model (SIOM). Because of the large scale and non-linear aspect of the problem, SIOM pre-generates candidate order-quantities for every NIIN, and then optimizes order-points,  $s$ , and order-quantities,  $Q$  (restricted to pre-generated ones). SIOM is implemented using the General Algebraic Modeling System (GAMS 2016a) with the GAMS CPLEX (GAMS 2016b) as solving engine. For completeness, the formulation of SIOM is given in the Appendix.

At the time of this research, SIOM is undergoing testing at NAVSUP WSS. Preliminary results are promising, albeit computational times are long due to the complexity of the resulting optimization model.

In this study, we introduce a new technique based on pre-generation of both  $s$  and  $Q$  candidates. That is, we proceed as in SIOM but limit the values of, not only  $Q$ , but also  $s$ , to a list of pre-generated candidate  $(s, Q)$  pairs. The new optimization model is referred to as SIOM with pre-generated  $s$ - $Q$  (SIOMsQ).

The motivation for enumerating  $(s, Q)$  pairs by pre-generation is two-fold: First, computational time of SIOMsQ is expected to be shorter due to using a restricted set of candidate  $(s, Q)$  pairs. Second, and possibly the most important, once  $(s, Q)$  pairs are generated, fill rates can be calculated (for each item and  $(s, Q)$  pair) using whichever method is deemed appropriate, such as closed-form equations, simulation, etc. For example, Roth (2016) has devised an accurate simulation of fill rates, which could potentially be used in SIOMsQ as input data. In SIOM, since the order-points are decision variables themselves, the possibilities to calculate fill rates are reduced to an approximation via a series of equations and constraints embedded in the model, making it substantially more complex. For comparison purposes, we only calculate the fill rates for

the pre-generated  $(s, Q)$  pairs, mimicking the values that would have been calculated if SIOM had been used.

#### **D. RESEARCH OBJECTIVES AND SCOPE**

The objective of this research is to develop an algorithm to generate reasonable  $(s, Q)$  pairs for every NIIN, and to create the SIOMsQ formulation to approximate the solution of the original SIOM faster. SIOMsQ seeks to strike a proper balance between solution quality and solvability of the problem. That is, SIOMsQ may relinquish solution quality for a faster solution.

This research assumes the existing SIOM release 1.2.2 is a valid approach to the site demand-based inventory problem.

Some distribution sites have two types of NIINs and they are treated as different sites. NMC stands for “Navy-Managed Consumables” and BP28 refers to “Navy Working Capital Fund Budget Program 28” (Salmeron and Craparo 2015). The SIOMsQ approach will be compared to SIOM in the following test cases already provided by NAVSUP:

1. Norfolk (NMC)
2. Norfolk (BP28)
3. Yokosuka (NMC)
4. Yokosuka (BP28)
5. Kings Bay (NMC)
6. Kings Bay (BP28)
7. San Diego (NMC)
8. Bangor (BP28)
9. Jacksonville (BP28)
10. Key West (BP28)

## II. METHODOLOGY

### A. SIOM CHARACTERISTICS

SIOM (Salmeron and Craparo 2015) minimizes the aggregate nonlinear penalty for not satisfying the required fill rates of each item. The model is non-separable by item due to constraints on total budget and maximum number of orders per month, each of which involves all NIINs in the problem.

Due to non-linear relationships between order quantities, fill rates and number of orders placed, SIOM pre-generates a list of candidate  $Q$ -values for every NIIN, which allows us to formulate SIOM as a mixed-integer linear optimization problem.

The order quantities (currently up to ten candidate  $Q$ -values for every NIIN) are generated using the following ideas:

- Set a reasonable lower and upper limit for  $Q$ , based on the data (average demand, lead time, shelf life, etc.) already provided by NAVSUP WSS, and
- Select  $Q = 1$  as a candidate value and then remaining order-quantities between the upper and lower bounds (inclusive) such that step sizes between two consecutive candidates are approximately the same.

After generation of order-quantity candidates, SIOM optimizes  $s$  and  $Q$  (as restricted) for all NIINs.

However, in some of the sites that have thousands of items, SIOM's difference between the upper and lower bounds of the solution are large even after 5 hours of computation.

To be able to obtain a faster solution, SIOM incorporates a grouping heuristic algorithm (GHA). In GHA, items in a certain distribution site are separated into smaller groups, constraints are adjusted proportionally and sub-problems are solved individually. At the end, objective values of sub-problems are summed and the approximate solution is achieved. For intermediate size sites (3,000–5,000 items), the grouping algorithm gives good results in a reasonable amount of time. In larger cases (5,000+ items), in order to

obtain an acceptable solution, the number of groups should be increased which potentially reduces the quality of the solution.

## **B. SIOMSQ DEVELOPMENT**

### **1. Definitions**

The development of SIOMsQ needs the following terms to be defined:

Shelf life ( $\bar{S}_i^L$ ): Shelf life for a given item (in months).

Maximum Stock Level (MSL): Inventory up-to-level for a given item (i.e., same as  $S_i = s_i + Q_i$ ).

Expected monthly demand ( $\hat{x}_i^M$ ): Expected monthly demand for a given item.

Allowance ( $\underline{S}_i$ ): Minimum MSL level required for a given item. If not specified, the default value is 0 for each item.

Maximum Months of Supply ( $\bar{S}$ ): A parameter that restricts the MSL of each item. For instance, if  $\bar{S}$  is 12, MSL of an item cannot exceed 12 times the expected monthly demand of the item.

Minimum Months of Supply ( $\underline{S}$ ): The parameter that sets a lower bound on MSL of each item. If  $\underline{S}$  is (for example 0.5), maximum stock level cannot be less than 0.5 times the expected monthly demand of the item.

Minimum order-point ( $\underline{s}_i$ ): Lower bound on  $s$  for a given item. Default is -1, which means that we would wait until there are no items on-hand and a new demand arrives to place an order of the item.

### **2. Generation of Parameters $s$ and $Q$**

Order-points and order-quantities are integer decision variables which can take potentially many values. To simplify the problem, SIOMsQ uses a reduced set of candidates for both order-points and order-quantities. For example, in order to generate

candidate  $(s, Q)$  pairs we can generate up to 10 candidate order-quantities for every NIIN, and then 10 candidate order-points for each order-quantity and NIIN.

**a. Generation of Order-Quantities**

To generate order-quantities, we preserve the same scheme as in SIOM (Salmeron Craparo 2016):

- “1” is always the first candidate  $Q$  for each item.
- Set lower bound on the remaining  $Q$  values ( $\underline{Q}_i$ ) as  $\max\{\underline{S} \times \hat{x}_i^M, 2\}$ .
- If shelf life of the item is less than  $\bar{S}$ , set upper bound on  $Q$  ( $\bar{Q}_i$ ) as  $\bar{S}_i^L \times \hat{x}_i^M$ . Otherwise, set  $\bar{Q}_i$  as  $\max\{\bar{S}_i, \bar{S} \times \hat{x}_i^M\}$ .
- Set  $\underline{Q}_i$  and  $\bar{Q}_i$  as the second and the last candidates, respectively.
- Select the remaining candidates between  $\underline{Q}_i$  and  $\bar{Q}_i$ , such that the differences between consecutive candidates are approximately the same (i.e., round to the nearest integer for fractional numbers).

**b. Generation of Order-Points**

In SIOMsQ, generation of each candidate  $s$  depends on already generated candidate order-quantities. Therefore, every candidate  $Q$  has its own set of order-points:

- “-1” and “0” are always the first two candidate order-points for each item. Candidate “-1” stands for backorders.
- Set lower bound on the remaining  $s$  values ( $\underline{s}_i$ ) as  $\max\{\underline{S}_i - Q_i, 1\}$ .
- Set upper bound on  $s$  ( $\tilde{s}_i$ ) as  $\min\{\bar{S}_i^L \times \hat{x}_i^M - Q_i, \bar{S} \times \hat{x}_i^M + Q_i\}$ .
- Set  $\underline{s}_i$  and  $\tilde{s}_i$  as the third and the last candidates, respectively.
- Select the remaining candidates between  $\underline{s}_i$  and  $\tilde{s}_i$ , such that the differences between consecutive candidates are approximately the same (i.e., round to the nearest integer for fractional numbers).

**C. MATHEMATICAL MODEL**

**1. Indices and Index Sets**

$i$  Item (i.e., NIIN), for  $i \in I$ .



$h$	Index of candidate $(s, Q)$ pairs for a given item $i$ , $h \in \{1, 2, \dots\}$ .
$m$	Index for the penalty segments of fill rates (e.g., $m \in M = \{1, 2, \dots, 5\}$ ).
$l$	Group of items, for $l \in L$ .
<b>2. Input Data and Parameters [Units]</b>	
$t_i$	Lead time for item $i$ . [quarters]
$\hat{x}_i$	Expected demand for item $i$ during the lead time. [units of issue/lead time]
$\bar{h}_i$	Number of candidate pairs (order-point and order-quantity) for item $i$ .
$\tilde{s}_{ih}$	$s$ -value (order-point) of the $h$ -th candidate for item $i$ . [units of issue]
$\tilde{Q}_{ih}$	$Q$ -value (order-quantity) of the $h$ -th candidate for item $i$ . [units of issue]
$\underline{s}_i$	Lower bound on order-point of item $i$ (default is -1). [units of issue]
$\bar{s}_i$	Upper bound on order-point of item $i$ . [units of issue]
$\underline{S}_i$	Allowance for item $i$ . [units of issue]
$\Delta^S$	1 if allowances are activated, and 0 otherwise.
$c_i$	Cost per unit of item $i$ . [\$/unit]
$r$	Maximum number of total expected orders per month for all items. [orders/month]
$\bar{S}_i^L$	Shelf life for item $i$ . [quarters]
$\bar{S}$	Maximum months of supply for any item. [months]
$\hat{s}_i^0$	Initial order-point used to enforce persistence for item $i$ . [units of issue]
$\delta_i^p$	Penalty for deviation from initial order-point for item $i$ . [unitless]
$\gamma^p$	Overall persistence weight. [unitless]

$\delta_i^{\bar{s}}$	Penalty for deviation of item $i$ above maximum months of supply. [unitless]
$\gamma^{\bar{s}}$	Overall weight for MSL deviations above maximum months of supply. [unitless]
$l_i$	Group of item $i$ . [unitless]
$\bar{f}_l$	Required fill rate for any item in group $l$ . [fraction, unitless]
$w_l$	Weight for achieving the required fill rate of any item in group $l$ . [weight units]

### 3. Decision Variables

$\Gamma_{ih}$	1 if candidate pair $h$ is selected for item $i$ , and 0 otherwise.
$Q_i$	Order-quantity for item $i$ . [units of issue]
$s_i$	Order-point for item $i$ . [units of issue]

### 4. Derived Data

$f_{ih}$	Fill rate of item $i$ for candidate pair $h$ . [fraction, unitless]
$\tilde{f}_{ih}$	Penalty for selecting candidate pair $h$ for item $i$ . [unitless]
$s_{ih}^+$	Deviation down with respect to initial order-point for item $i$ and candidate pair $h$ . [units of issue]
$s_{ih}^-$	Deviation up with respect to initial order-point for item $i$ and candidate pair $h$ . [units of issue]
$\bar{S}_{ih}^+$	Deviation for MSL of item $i$ and candidate $h$ above the average demand during the maximum months of supply for the item. [months]

- $\tilde{c}_{ih}$  Expected number of cycles per lead time period for item  $i$  if candidate pair  $h$  is selected. [unitless] Note: A cycle is defined as the time period between consecutive orders, or the lead time; whichever is shorter.
- $\tilde{B}_{ih}$  Expected fraction of backorders during a cycle for item  $i$  if candidate pair  $h$  is selected. [fraction, unitless] Note: The calculation method appears in Equation (1).
- $X'_{ih}$  Random variable for the demand of item  $i$  in a cycle if candidate pair  $h$  is selected. [units of issue].
- $f_{ih}^{dev}$  Deviation from the required fill rate for item  $i$  if candidate pair  $h$  is selected. [fraction, unitless]
- $\tilde{f}_{ihm}^{dev}$  Deviation penalty for item  $i$ , in fill rate penalty segment  $m$  if candidate pair  $h$  is selected. [unitless]
- $\bar{f}_{im}^-$  Maximum deviation allowed for fill rate of item  $i$ , in penalty segment  $m$ .  
(Calculated as  $\bar{f}_{im}^- = f_i \frac{m^2}{\sum_{j \in M} j^2}$ ) [fraction, unitless]
- $w_{l,m}$  Penalty for deviation from the required fill rate for items of group  $l$  in penalty segment  $m$ . (calculated as  $w_{l,m} = mw_l$ ) [unitless]

After generating all the candidate  $(s, Q)$  pairs for all items, we can pre-calculate the fill rate of each item for each candidate pair,  $h = (s, Q)$ .

The only reason that could cause fill rate to be less than 100% is backorders. Therefore, in order to calculate the fill rate of every item for every corresponding  $h$ , first we need to compute the expected fraction of backorders,  $\tilde{B}_{ih}$ , during a cycle (see definition of  $\tilde{c}_{ih}$ ) for each item and for every  $h$ . The expected fraction of backorders can be calculated using the following formula:

$$\tilde{B}_{ih} = \frac{1}{\tilde{Q}_{ih}} \int_{s'_{ih}}^{\infty} (x - s'_{ih}) f_{X'_{ih}}(x) dx \quad (1)$$

In Equation (1),  $X'_{ih}$  stands for the random variable for the demand in a cycle with probability distribution given by density function (or mass function)  $f_{X'_{ih}}(x)$  and  $s'_{ih} = s_{ih} - (\tilde{c}_{ih} - 1)Q_{ih}$ , where  $\tilde{c}_{ih} = \max\left\{1, \frac{\hat{x}_i}{Q_{ih}}\right\}$ .

After calculation of expected fraction of backorders, fill rate of each NIIN can be computed (for every  $h$ ) as follows:  $f_{ih} = 1 - \tilde{B}_{ih}$ . This formula can return  $f_{ih} < 0$  in which case we use  $f_{ih} = 0$ .

Penalties are applied for the items that cannot achieve the required fill rate. For every site, there are certain groups of items that all have the same required fill rates. Deviation from the required fill rate is formulated as  $f_{ih}^{dev} = \max\{0, \bar{f}_{l_i} - f_{ih}\}$ , where  $\bar{f}_{l_i}$  is the required fill rate for item  $i$  of group  $l$ .

Before calculating the total penalties for fill rate deviations, we need to compute the deviation penalties in every penalty segment for each item,  $\tilde{f}_{ihm}^{dev}$ , which is calculated as follows:

$$\tilde{f}_{ihm}^{dev} = \begin{cases} \bar{f}_{im}^- w_{l,m} & \text{if } f_{ih}^{dev} > \sum_{m' \leq m} \bar{f}_{im'}^- \\ \left( f_{ih}^{dev} - \sum_{m' \leq m-1} \bar{f}_{im'}^- \right) w_{l,m} & \text{if } \sum_{m' \leq m-1} \bar{f}_{im'}^- < f_{ih}^{dev} \leq \sum_{m' \leq m} \bar{f}_{im'}^- \\ 0 & \text{if } f_{ih}^{dev} \leq \sum_{m' \leq m-1} \bar{f}_{im'}^- \end{cases}$$

Finally, the deviation penalty for selecting candidate pair  $h$  of item  $i$  can be calculated as  $\tilde{f}_{ih} = w_{l_i} \sum_m \tilde{f}_{ihm}^{dev}$ .

Another derivation that has to be computed as an input to SIOMsQ is  $s_{ih}^+$  and  $s_{ih}^-$ . They are calculated as  $s_{ih}^+ = \max\{\hat{s}_i^0 - \tilde{s}_{ih}, 0\}$  and  $s_{ih}^- = \max\{\tilde{s}_{ih} - \hat{s}_i^0, 0\}$ .

Lastly, in order to calculate MSL deviation penalty,  $\bar{s}_{ih}^+$  has to be derived as

$$\bar{s}_{ih}^+ = \max \left\{ \frac{3t_i}{\hat{x}_i} (\tilde{s}_{ih} + \tilde{Q}_{ih}) - \bar{S}, 0 \right\}.$$

## 5. SIOMsQ Formulation

SIOMsQ can be stated as the following mixed-integer, linear problem:

$$\text{SIOMsQ: } \min_{s, Q, \Gamma} \sum_i \sum_{h|h \leq \bar{h}_i} \tilde{f}_{ih} \Gamma_{ih} + \gamma^p \sum_i \sum_{h|h \leq \bar{h}_i} \frac{\delta_i^p}{\hat{s}_i^0 + 1.5} (s_{ih}^+ + s_{ih}^-) \Gamma_{ih}^h + \gamma^{\bar{s}} \sum_i \sum_{h|h \leq \bar{h}_i} \frac{\delta_i^{\bar{s}}}{\bar{S} + 1} \bar{s}_{ih}^+ \Gamma_{ih}^h \quad (2)$$

Subject to:

$$Q_i = \sum_{h|h \leq \bar{h}_i} \tilde{Q}_{ih} \Gamma_{ih} \quad \forall i \quad (3)$$

$$s_i = \sum_{h|h \leq \bar{h}_i} \tilde{s}_{ih} \Gamma_{ih} \quad \forall i \quad (4)$$

$$\sum_{h|h \leq \bar{h}_i} \Gamma_{ih} = 1 \quad \forall i \quad (5)$$

$$\sum_i c_i (s_i + Q_i - \lceil \underline{S}_i \rceil \Delta^S) \leq b \quad (6)$$

$$\sum_i \sum_{h|h \leq \bar{h}_i} \frac{\hat{x}_i / \tilde{Q}_{ih}}{3t_i} \Gamma_{ih} \leq r \quad (7)$$

$$\lceil \underline{S}_i \rceil \Delta^S \leq s_i + Q_i \leq \bar{S}_i^L \frac{\hat{x}_i}{3t_i} \quad \forall i \quad (8)$$

$$\underline{S}_i \leq s_i \leq \bar{S}_i \quad \forall i \quad (9)$$

$$Q_i \geq 0 \text{ and integer} \quad \forall i \quad (10)$$

$$s_i \geq -1 \text{ and integer} \quad \forall i \quad (11)$$

$$\Gamma_{ih} \in \{0, 1\} \quad \forall i, h \mid h \leq \bar{h}_i \quad (12)$$

The goals of the objective function (2) are to minimize (a) deviations from target fill rates for all items (larger penalty rates for being far away from target fill rate); (b) the penalties for deviating from current order-points; and (c) the penalties for exceeding maximum months of supply. Persistence penalties and MSL deviation penalties are optional and can be removed from the objective function by setting  $\gamma^p = 0$  and  $\gamma^{\bar{s}} = 0$ , respectively.

Equations (3) – (5) restrict the model in choosing only one  $(s, Q)$  pair for an item and set the values of selected  $s$  and  $Q$ .

Equation (6) establishes a budget restriction on the MSL cost of all items. Equation (7) restricts the expected number of orders placed in a month from exceeding a given limit.

Equation (8) limits the lower and upper bounds of MSL. Stock levels should be greater than the allowance and less than the minimum amount that would cause items to perish (under expected demand assumptions).

Equations (9) – (12) set the domains of the decision variables and parameters.

## 6. Two-Step Approach

While running SIOMsQ (regular method), we empirically found that it takes substantial time for GAMS/CPLEX to identify an initial solution before starting branch-and-bound. In fact, in rare occasions, a feasible solution (i.e., integer) cannot be found in the allotted time of two hours. In order to speed up SIOMsQ solution, we have developed the following two-step approach:

- In the first step, we solve a further restricted SIOMsQ, called SIOMsQ'. This uses only a random subset from the original candidate ( $s$ ,  $Q$ ) pairs, for each NIIN. SIOMsQ' shortens the time for an initial solution calculation. We do not fully solve SIOMsQ': we stop after the first feasible solution has been identified. We denote by  $\bar{h}_i' \leq \bar{h}_i$ , the number of candidate ( $s$ ,  $Q$ ) pairs for item  $i$  used in SIOMsQ'.
- In the second step, we run the full SIOMsQ (with all candidate pairs) using the initial solution found by SIOMsQ'.

## D. COMPUTATIONAL EXPERIENCE

Both SIOM and SIOMsQ are executed on a Lenovo W520 laptop computer with 16 GB of RAM and two 2.5 GHz Intel Core i7 processors. We direct GAMS/CPLEX to use up to four parallel threads, which speeds up calculations by a factor of 2.0 in some cases. All SIOMsQ cases are solved to within 1% of optimality or using two hours of computational time, whichever comes first. SIOMsQ' is solved until the first feasible solution is identified.

THIS PAGE INTENTIONALLY LEFT BLANK

### III. COMPUTATIONAL ANALYSIS

We run SIOMsQ for ten cases provided by NAVSUP. We present the results of these runs with various options and compare them to SIOM.

#### A. RUNNING OPTIONS FOR SIOM AND SIOMSQ

SIOM is executed with time limit of 2 hours or 5 hours (if 2 hours is not enough to decrease the optimality relative tolerance to 1%) for each site. SIOMsQ always has up to 2 hours of running time. For the two-step approach, the time spent in SIOMsQ' is not counted in the two-hour limit allowed for the full SIOMsQ. Additional time is allowed for pre-processing of the data.

Apart from full group size, SIOM may incorporate a grouping heuristic in which items are grouped into either 500 or 1,000 NIINs in order to solve the problem faster. The grouping algorithm is used only for the cases where an acceptable solution is not achieved within given time limit. SIOMsQ is always run with all NIINs.

Both SIOM and SIOMsQ use 1% optimality relative tolerance. That is, while branch-and-bound is being executed, when the relative difference between the incumbent objective function value and the best possible value of the solution (lower bound) drops below 1%, the solver stops and the incumbent solution is reported.

For a given set of pre-generated  $Q$ -values, lower bound on the objective function value is only guaranteed to be valid for SIOM if run with no grouping of NIINs. Lower bounds provided by SIOM with a grouping heuristic and by SIOMsQ are only estimates.

SIOM pre-generates  $\#Q = 10$  candidate order-quantities and fully optimizes order-points for each item. In SIOMsQ, however, both  $s$  and  $Q$  candidates are pre-generated. In order to see the effects of number of pre-generated order-points and order-quantities on the solution quality and running time, we use various sets of  $(s, Q)$  pairs. We denote by  $(\#s, \#Q)$  a specific run where we pre-generate  $\#Q$  candidates for  $Q$  for each NIIN, and then  $\#s$  candidates for  $s$  for each of the  $Q$  candidates. We test the



following combinations of  $(\#s, \#Q)$ : (7, 7), (7, 10), (10, 10), (15, 10), (15, 15) and (20, 10).

SIOMsQ is run with the regular method (in one step without pre-generating a feasible initial solution), and with two-step approach (described in Section II.C). In the latter case, we use SIOMsQ' with a reduced number of candidate pairs  $\bar{h}_i' = 50 \forall i$  in order to generate an initial solution faster.

## B. RESULTS

### 1. Yokosuka (NMC)

Yokosuka (NMC) is the smallest case with 35 NIINs. SIOM solves the problem in less than a minute with no penalty applied to the objective function value. That is, the required fill rates are satisfied optimally with the solution suggested by SIOM for each item. (See Table 1.)

Table 1. Yokosuka (NMC) SIOM Results

Time Limit	Group Size	Objective	Lower Bound	Run Time (mins)
2 hours	full	0	0	<1

SIOMsQ also solves the problem in less than a minute for all possible solving options. However, when the number of  $s$  and  $Q$  candidates are (7, 7) and (7, 10) the objective function value is slightly sub-optimal. The regular method and the two-step approach do not make any difference in the solution for this case. (See Table 2.)

For this case, the SIOMsQ optimization problem size ranges from 5,217 integer variables and 248 single equations in the (7, 7) case to 5,708 integer variables and 248 single equations in the (15, 15) case.

Table 2. Yokosuka (NMC) SIOMsQ Results

# $s$	# $Q$	Method	Objective	Estimated Lower Bound	Run Time (mins)
7	7	Regular	0.008	0.008	<1
		Two-step	0.008	0.008	<1
7	10	Regular	0.008	0.008	<1
		Two-step	0.008	0.008	<1
10	10	Regular	0	0	<1
		Two-step	0	0	<1
15	10	Regular	0	0	<1
		Two-step	0	0	<1
15	15	Regular	0	0	<1
		Two-step	0	0	<1
20	10	Regular	0	0	<1
		Two-step	0	0	<1

## 2. Key West (BP28)

Key West (BP28) is one of the smallest test cases with 53 items. SIOM gives a solution of 1.40 in less than a minute. (See Table 3.)

Table 3. Key West (BP28) SIOM Results

Time Limit	Group Size	Objective	Lower Bound	Run Time (mins)
2 hours	full	1.40	1.39	<1

The running time of SIOMsQ for Key West (BP28) site is also less than a minute. When it is run with 7 candidates of  $s$  and  $Q$  each, the solution is worse than all the other cases. This is an expected outcome, since coverage in  $s$  and  $Q$  is low.

In the case of 15 candidates for both  $s$  and  $Q$ , the solution is even better than the lower bound suggested by SIOM. The reason is that SIOM uses 10 pre-generated order-quantities. Due to a better coverage in  $Q$ , the solution quality can potentially be better in the (15, 15) case. (See Table 4.)

Table 4. Key West (BP28) SIOMsQ Results

# $s$	# $Q$	Method	Objective	Estimated Lower Bound	Run Time (mins)
7	7	Regular	1.47	1.47	<1
		Two-step	1.47	1.47	<1
7	10	Regular	1.40	1.40	<1
		Two-step	1.41	1.40	<1
10	10	Regular	1.40	1.40	<1
		Two-step	1.40	1.40	<1
15	10	Regular	1.40	1.40	<1
		Two-step	1.40	1.40	<1
15	15	Regular	1.38	1.37	<1
		Two-step	1.38	1.37	<1
20	10	Regular	1.40	1.40	<1
		Two-step	1.40	1.40	<1

### 3. Kings Bay (NMC)

There are 251 items in Kings Bay (NMC). SIOM achieves 30.7 as the objective function value in less than a minute with the full group size. (See Table 5.)

Table 5. Kings Bay (NMC) SIOM Results

Time Limit	Group Size	Objective	Lower Bound	Run Time (mins)
2 hours	full	30.7	30.7	<1

When we run SIOMsQ with various sets of  $(s, Q)$  pairs, we always get the same objective function value of 30.7. Using regular method or two-step approach does not change the solution. (See Table 6.)

Table 6. Kings Bay (NMC) SIOMsQ Results

# $s$	# $Q$	Method	Objective	Estimated Lower Bound	Run Time (mins)
7	7	Regular	30.7	30.7	<1
		Two-step	30.7	30.7	<1
7	10	Regular	30.7	30.7	<1
		Two-step	30.7	30.7	<1
10	10	Regular	30.7	30.7	<1
		Two-step	30.7	30.7	<1
15	10	Regular	30.7	30.7	<1
		Two-step	30.7	30.7	<1
15	15	Regular	30.7	30.7	<1
		Two-step	30.7	30.7	<1
20	10	Regular	30.7	30.7	<1
		Two-step	30.7	30.7	<1

#### 4. San Diego (NMC)

San Diego (NMC) has 468 NIINs in its inventory. SIOM solves the problem in 19 minutes and the objective function value is 103. (See Table 7.)

Table 7. San Diego (NMC) SIOM Results

Time Limit	Group Size	Objective	Lower Bound	Run Time (mins)
2 hours	full	103	102	19

We obtain approximately the same objective function value as the solution in SIOM when we run SIOMsQ for any number of candidate  $s$  and  $Q$  or solution method, but it takes less than a minute in all cases. (See Table 8.)

Table 8. San Diego (NMC) SIOMsQ Results

# $s$	# $Q$	Method	Objective	Estimated Lower Bound	Run Time (mins)
7	7	Regular	104	103	<1
		Two-step	104	103	<1
7	10	Regular	104	103	<1
		Two-step	103	103	<1
10	10	Regular	102	102	<1
		Two-step	102	102	<1
15	10	Regular	103	102	<1
		Two-step	103	102	<1
15	15	Regular	103	102	<1
		Two-step	103	102	<1
20	10	Regular	103	102	<1
		Two-step	103	102	<1

## 5. Norfolk (NMC)

Norfolk (NMC) is a mid-size case with 1,189 items. We run SIOM using three methods: full, grouping by 500 and grouping by 1,000. (Note: grouping is done in the sequential order that the NIINs appear in the input file.) All the methods give the same objective function value of 212. The full model takes 92 minutes to run whereas the grouping methods requires much less time (25 and 26 minutes respectively). (See Table 9.)

Table 9. Norfolk (NMC) SIOM Results

Time Limit	Group Size	Objective	Lower Bound	Run Time (mins)
2 hours	500	212	211*	25
	1,000	212	211*	26
	full	212	209	92

\* Estimated, not necessarily valid.

The objective function values suggested by SIOMsQ are slightly above that of SIOM for all  $(s, Q)$  sets, but the run times of SIOMsQ are much smaller. (See Table 10.)

Table 10. Norfolk (NMC) SIOMsQ Results

# $s$	# $Q$	Method	Objective	Estimated Lower Bound	Run Time (mins)
7	7	Regular	216	215	<1
		Two-step	216	215	<1
7	10	Regular	216	215	1
		Two-step	217	215	1
10	10	Regular	215	215	<1
		Two-step	215	215	1
15	10	Regular	215	215	2
		Two-step	215	215	<1
15	15	Regular	216	215	1
		Two-step	215	215	3
20	10	Regular	215	215	2
		Two-step	215	215	1

## 6. Jacksonville (BP28)

There are 3,256 NIINs in the inventory of Jacksonville (BP28). SIOM is run as full and with grouping method restricted to 2 hours and 5 hours for each method. In none of these cases is the optimality gap reduced below 1%. The model with time limit of 5 hours gives the best objective solution (37). (See Table 11.)

Table 11. Jacksonville (BP28) SIOM Results

Time Limit	Group Size	Objective	Lower Bound	Run Time (mins)
2 hours	500	39	30*	122
	1,000	46	26*	122
	full	121	25	122
5 hours	500	37	32*	302
	1,000	37	27*	302
	full	38	25	302

\* Estimated, not necessarily valid.

When we run SIOMsQ, we achieve very reasonable objective function values in much shorter time comparing to SIOM. The run time of the (20, 10) case is only 3 minutes and the achieved objective function value is better than SIOM.

As expected, as we increase the number of candidate ( $s$ ,  $Q$ ) pairs, we observe lower objective function values. Both the regular and two-step methods do not make any difference in this case. (See Table 12.)

Table 12. Jacksonville (BP28) SIOMsQ Results

# $s$	# $Q$	Method	Objective	Estimated Lower Bound	Run Time (mins)
7	7	Regular	41	41	1
		Two-step	41	41	1
7	10	Regular	40	40	1
		Two-step	40	40	1
10	10	Regular	36	36	2
		Two-step	36	36	2
15	10	Regular	34	34	2
		Two-step	34	34	2
15	15	Regular	34	34	3
		Two-step	34	34	3
20	10	Regular	33	33	3
		Two-step	33	33	3

## 7. Kings Bay (BP28)

Kings Bay (BP28) has 4,513 NIINs in its inventory. A full and grouped SIOM give almost the same objective function value but the running time of grouping heuristics are much shorter than the full SIOM. (See Table 13.)

Table 13. Kings Bay (BP28) SIOM Results

Time Limit	Group Size	Objective	Lower Bound	Run Time (mins)
2 hours	500	132	132*	34
	1,000	133	132*	34
	full	132	131	121

\* Estimated, not necessarily valid.

While running SIOMsQ, and for any ( $s$ ,  $Q$ ) pairs generated, the objective function value does not change. It is barely greater than the result of SIOM, but run times are shorter.

For (15, 15) and (20, 10) cases, executing regular SIOMsQ takes more time than the two-step SIOMsQ. The main reason for this is that the calculation of an initial solution takes very long without using the restricted SIOMsQ'. (See Table 14.)

Table 14. Kings Bay (BP28) SIOMsQ Results

# $s$	# $Q$	Method	Objective	Estimated Lower Bound	Run Time (mins)
7	7	Regular	134	134	3
		Two-step	134	134	3
7	10	Regular	134	134	4
		Two-step	134	134	5
10	10	Regular	134	134	7
		Two-step	134	134	4
15	10	Regular	134	134	4
		Two-step	134	134	4
15	15	Regular	134	134	29
		Two-step	134	134	6
20	10	Regular	134	134	10
		Two-step	134	134	6

## 8. Norfolk (BP28)

Norfolk (BP28) is a large test case with 7,556 items. When full SIOM is run with 2 and 5 hours limit, the suggested solutions are not acceptable since the differences between achieved objective function value and the lower bound are too big. When a grouping heuristic is used, the results are much better for both 2-hour and 5-hour cases, but still not good enough to determine the answer to the problem. (See Table 15.)

Table 15. Norfolk (BP28) SIOM Results

Time Limit	Group Size	Objective	Lower Bound	Run Time (mins)
2 hours	500	354	64*	127
	1,000	348	60*	127
	full	17,533	59	128
5 hours	500	170	70*	307
	1,000	194	61*	307
	full	13,423	59	308

\* Estimated, not necessarily valid.



The solutions given by SIOMsQ are better than those by SIOM, since the objective function values are close to the provable lower bound of SIOM, and run times are acceptable even for larger number of candidates (10 – 20 minutes). The results suggested by the (15, 10), (15, 15) and (20, 10) cases are better than those suggested by other cases. (See Table 16.)

Table 16. Norfolk (BP28) SIOMsQ Results

# $s$	# $Q$	Method	Objective	Estimated Lower Bound	Run Time (mins)
7	7	Regular	138	138	6
		Two-step	138	138	6
7	10	Regular	130	130	8
		Two-step	130	130	9
10	10	Regular	99	99	12
		Two-step	99	99	10
15	10	Regular	84	84	12
		Two-step	84	84	12
15	15	Regular	84	83	20
		Two-step	84	83	15
20	10	Regular	80	80	14
		Two-step	80	80	13

## 9. Bangor (BP28)

Bangor (BP28) is the second largest test case with an inventory that consists of 8,141 NIINs. When full SIOM is run, it cannot reduce the optimality relative tolerance to 1% in the given time limits (2 or 5 hours). The result of full SIOM is not even close to the optimal solution, but using the grouping heuristic gives reasonable approximations (an objective function value of 287 in the best case, for a lower bound of 251) especially when we allow it to run for 5 hours. (See Table 17.)

Table 17. Bangor (BP28) SIOM Results

Time Limit	Group Size	Objective	Lower Bound	Run Time (mins)
2 hours	500	361	272*	129
	1,000	390	258*	129
	full	10,219	251	129
5 hours	500	292	276*	309
	1,000	287	263*	309
	full	6,104	251	309

\* Estimated, not necessarily valid.

The objective function values suggested by SIOMsQ are acceptable (between 269 and 271), improving SIOM, and running times are shorter. The number of  $(s, Q)$  pairs or the two-step method does not make a significant difference in solution quality for this case. (See Table 18.)

Table 18. Bangor (BP28) SIOMsQ Results

# $s$	# $Q$	Method	Objective	Estimated Lower Bound	Run Time (mins)
7	7	Regular	270	270	8
		Two-step	270	270	8
7	10	Regular	271	270	12
		Two-step	271	270	14
10	10	Regular	269	269	36
		Two-step	269	269	28
15	10	Regular	270	269	15
		Two-step	270	269	14
15	15	Regular	271	269	32
		Two-step	271	269	32
20	10	Regular	271	269	17
		Two-step	271	269	17

## 10. Yokosuka (BP28)

There are 11,798 items in the Yokosuka (BP28) test case. Due to its large scale, the SIOM solution quality is not good enough since the optimality relative tolerance cannot be reduced below 1% even if grouping heuristic is used. The best solution is achieved for the 5 hour - 500 group size run with an objective function value of 4,211 and

a (provable) lower bound of 3,321 (i.e., a potential solution gap of at most 26.8%). (See Table 19.)

Table 19. Yokosuka (BP28) SIOM Results

Time Limit	Group Size	Objective	Lower Bound	Run Time (mins)
2 hours	500	5,093	3,393*	138
	1,000	15,335	3,391*	137
	full	45,820	3,310	137
5 hours	500	4,211	3,433*	318
	1,000	8,191	3,399*	318
	full	32,401	3,321	322

\* Estimated, not necessarily valid.

When we run SIOMsQ, we obtain acceptable solutions in a reasonable amount of time. Specifically, we obtain an objective function value under 3,700 in all runs, which means much smaller solution gaps (always under 11.4%). This calculation uses the provable lower bound of SIOM, given SIOMsQ's bound is only an estimate. (See Table 20.)

There are 2,041,029 integer variables and 82,589 single equations in SIOMsQ run for the (15, 15) case in Yokosuka (BP28).

Table 20. Yokosuka (BP28) SIOMsQ Results

# $s$	# $Q$	Method	Objective	Lower Bound	Run Time (mins)
7	7	Regular	3,686	3,661	30
		Two-step	3,686	3,661	30
7	10	Regular	3,661	3,661	118
		Two-step	3,691	3,661	58
10	10	Regular	3,664	3,660	81
		Two-step	3,665*	3,660*	97*
15	10	Regular	3,659	3,659	93
		Two-step	3,659	3,659	91
15	15	Regular	3,661	3,659	171
		Two-step	3,660	3,659	173
20	10	Regular	3,684	3,658	135
		Two-step	3,663	3,658	140

\* For  $\bar{h}_i' = 30$  (instead of  $\bar{h}_i' = 50$  used in all other cases)

### C. SUMMARY OF RESULTS

The results of the all test cases are summarized in Table 21. We present the best solutions achieved by SIOM along with its lower bounds and running times for each test case. We also show the solutions, candidate pair sizes, execution times and the solution methods of best SIOMsQ for every case.

For all of the test cases, the solutions that SIOMsQ achieves are better than or very close to that of SIOM, but the running times of SIOMsQ are substantially shorter. As the size of the test cases increases, the best SIOMsQ solutions are achieved with the large candidate sizes such as (10, 10), (15, 15) or (20, 10). The two-step approach (including the time spent in solving the first step model, SIOMsQ') is faster especially in the larger test cases.

Table 21. Results Summary of Test Cases

Test Case	Best SIOM Solution	Provable SIOM Lower Bound	SIOM Time (mins)	Best SIOMsQ (#s, #Q)	Best SIOMsQ Solution	Best SIOMsQ Time (mins)	Best SIOMsQ Method
Yokosuka (NMC)	0	0	<1	Multiple	0	<1	Both
Key West (BP28)	1.40	1.39	<1	(15, 15)	1.38*	<1	Both
Kings Bay (NMC)	30.7	30.7	<1	Multiple	30.7	<1	Both
San Diego (NMC)	103	102	19	(10, 10)	102	<1	Both
Norfolk (NMC)	212	209	92	Multiple	215	<1	Both
Jacksonville (BP28)	37	25	302	(20, 10)	33	3	Both
Kings Bay (BP28)	132	131	121	Multiple	134	3	Both
Norfolk (BP28)	170	59	307	(20, 10)	80	13	Two-step
Bangor (BP28)	287	251	309	(10, 10)	269	28	Two-step
Yokosuka (BP28)	4,211	3,321	318	(15, 10)	3,659	91	Two-step

\* Due to a better coverage in  $Q$ , the optimal solution by SIOMsQ in the (15, 15) case can potentially be better than that of SIOM.

## IV. CONCLUSIONS AND FUTURE RESEARCH

### A. CONCLUSIONS

In this thesis, we have developed SIOMsQ, a new approach to solving SIOM (a mathematical program developed at the Naval Postgraduate School to guide NAVSUP's inventory levels of site demand-based items).

We have successfully implemented SIOMsQ, which restricts SIOM by reducing the number of candidate order-points in exchange for more efficient solutions. The approach creates an educated specification of a list of candidate  $(s, Q)$  pairs and solves the optimization problem using only those candidates as decision variables. SIOMsQ may still be difficult to solve in some cases, so we further restrict it using a two-step approach in order to find an initial solution with which to speed SIOMsQ up.

We run 10 test cases provided by NAVSUP with various combinations of candidate pair sizes. In eight of ten cases, we obtain at least as good solutions as SIOM in the same or substantially less time. In the two other cases, the differences are minimal, and computational times are still favorable.

As a conclusion of this research, we recommend that running SIOMsQ using  $(\#s, \#Q)$  values of (20, 10) or (7, 7), and a two-step approach are the preferred options. (7, 7) cases give good-enough results faster than (20, 10) cases, but the solutions suggested by (20, 10) cases are more precise. Therefore, there is a tradeoff between solution quality and solving time, although in most cases, differences are also small. We recommend using SIOMsQ versus the current SIOM.

### B. FUTURE RESEARCH

As we have successfully implemented SIOMsQ, further studies can focus on:

- Having SIOMsQ fully tested by NAVSUP,
- Integrating SIOMsQ as an option for NAVSUP to use from the graphical interface of SIOM,

- Analyzing the optimal configuration of  $(\#s, \#Q)$  pairs for each individual test case, or even developing a methodology that allows SIOMsQ to dynamically accommodate different candidate pair sizes.
- Speeding up pre-processing of generating  $(s, Q)$  candidate pairs, the probability distributions of demand based on  $Q$ , and derived data based on  $s$  and  $Q$  (all of which takes up to 50 minutes in some cases).

## APPENDIX. SIOM FORMULATION

This appendix describes the formulation of SIOM as given in Salmeron and Craparo (2016).

### A. INDICES AND INDEX SETS

- $i$  ,      item (i.e., NIIN), for  $i \in I$ .
- $l$  ,      Group. (Note:  $l$  here could also represent a combination of group and other criteria, such as group A, B, C, D, for the purpose of establishing different target fill rates.)
- $q$  ,      index for candidate order quantities, for  $q \in \{1, \dots, \bar{q}_i\}$  (see  $\bar{q}_i$  below).
- $n$  ,      demand-level index, for  $n \in N$  (e.g.,  $N = \{1, 2, \dots, 10\}$  represents ten levels of demand). This is actually item-dependent, and order size dependent, see  $n_i^q$  parameter below.
- $m$  ,      penalty segment, for  $m \in M$  (e.g.,  $M = \{1, 2, \dots, 5\}$  represents five levels of penalties for deviations, with respect to desired fill rate levels).

### B. INPUT DATA AND PARAMETERS

- $t_i$  ,      lead time length for item  $i$  [quarters/lead time].
- $\hat{x}_i, \hat{\sigma}_i$  , expected demand and estimated deviation, respectively, for item  $i$  during the lead time [items/lead time], [items/lead time].
- $\bar{q}_i$  ,      number of candidate order quantities for item  $i$ . This number depends on  $\hat{x}_i$ ,  $\bar{S}_i^L$ ,  $\bar{S}$ , and  $\underline{S}_i$ .
- $\tilde{Q}_i^q$  ,       $q$ -th candidate order quantity for item  $i$  [units/order].



$\tilde{c}_i^q$ , cycles during a lead time for item  $i$  and  $q$ -th candidate order quantity.

Calculated as  $\tilde{c}_i^q = \max\{1, \frac{\hat{x}_i}{\tilde{Q}_i^q}\}$  [orders per lead time].

$\Delta_i^{\tilde{c}q}$ , calculated as: one if  $\hat{x}_i + 1 > \tilde{Q}_i^q$ , and zero otherwise.

$\bar{\Delta}_i^{\tilde{c}}$ , calculated as:  $\bar{\Delta}_i^{\tilde{c}} = \max_{q|q \leq \tilde{q}_i} \{\Delta_i^{\tilde{c}q}\}$ .

$\hat{x}_i'^q, \hat{\sigma}_i'^q$ , expected demand and estimated deviation adjusted for cycle time, for  $q$ -th candidate order quantity for item  $i$ . Calculated as:  $\hat{x}_i'^q = \hat{x}_i / \tilde{c}_i^q$  and  $\hat{\sigma}_i'^q = \hat{\sigma}_i' / \tilde{c}_i^q$ .

$\bar{f}_l$ , desired (target) fill rate for each item in group  $l$  [fraction].

$w_l$ , weight for meeting required fill rate each item in group  $l$  [weight units].

$w_{lm}$ , penalty for deviating from required fill rate for items in group  $l$  within penalty segment  $m$ . Calculated as  $w_{lm} = m^e w_l$ , where  $e$  is a user input (penalty exponent, default  $e = 1$ ).

$\bar{f}_{im}^-$ , maximum deviation allowed for fill rate for item  $i$  within penalty segment  $m$ .

Calculated as  $\bar{f}_{im}^- = \bar{f}_{l_i} \frac{m^2}{\sum_{j \in M} j^2}$  where  $i \in I_l$ .

$OL_i$ , operating level for NIIN  $i$  [months].

$SL_i$ , safety level for NIIN  $i$  [months].

$\underline{S}_i$ , allowance for item  $i$  [demand units].

$\Delta^S$ , one if allowances are activated, and zero otherwise.

$c_i$ , cost per unit at MSL [\$ / unit].

$b$ , overall budget [\$]. Calculated as:

$$b = \sum_i c_i \left( \max \left\{ 0, \frac{(OL_i + SL_i) \hat{x}_i^{\text{QT}}}{3} - \underline{s}_i \right\} + \hat{x}_i \right), \text{ if allowances are activated } (\Delta^S = 1), \text{ or}$$

$$b = \sum_i c_i \left( \max \left\{ 0, \frac{(OL_i + SL_i) \hat{x}_i^{\text{QT}}}{3} \right\} + \hat{x}_i \right), \text{ otherwise } (\Delta^S = 0),$$

where  $\hat{x}_i^{\text{QT}}$  = average quarterly demand for NIIN  $i$  [units/quarter], derived as

$$\hat{x}_i^{\text{QT}} = \hat{x}_i / t_i .$$

$p_{in}^q, d_{in}^q$ ,  $n$ -th level of probability and demand, respectively, adjusted for the cycle time for the  $q$ -th candidate order quantity for item  $i$  [unitless], [demand units]. Note: If probability distributions are estimated based on mean and variance, use  $\hat{x}_i'^q$  and  $(\hat{\sigma}_i'^q)^2$  to generate these.

$n_i^q$ , number of demand levels, adjusted for the cycle time for the  $q$ -th candidate order quantity for item  $i$ .

$\bar{d}_i^q$  derived data:  $\bar{d}_i^q = \max_{n \leq n_i^q} \{\bar{d}_{in}^q\}$ ; [demand units].

$r$ , maximum number of total expected orders per month [orders]. Note: Can be fractional.

$\underline{s}_i, \bar{s}_i$ , lower and upper bounds on reorder point for item  $i$  [items], [items].

$$\underline{s}_i \geq -1 .$$

$\bar{S}_i^L$ , shelf life for item  $i$  (to establish its MSL upper bound) [months].

$\bar{S}$ , maximum months of supply for any item (to establish its MSL upper bound) [months].

$\hat{s}_i^0$ , initial reorder point used to enforce persistence for item  $i$  [items].

$\delta_i^P$ , penalty for deviation from initial reorder point for item  $i$  [weight units].

- $\gamma^p$ , overall persistence level [weight units].
- $\delta_i^{\bar{S}}$ , penalty for item's  $i$  deviation above maximum months of supply [weight units].
- $\gamma^{\bar{S}}$ , overall weight for MSL deviations above maximum months of supply [weight units].
- $\tilde{M}_i^q$ , large number greater (in magnitude) than any possible 'negative fill rate' value given by the original fill rate formula for the  $q$ -th candidate order quantity for item  $i$ , if  $\Delta_i^{\tilde{c}^q} = 1$ . For example, we may use  $\tilde{M}_i^q = 1,000$ . However, we can estimate a better value for numerics:  

$$\tilde{M}_i^q = \left| 2 - \hat{x}_i'^q / \tilde{Q}_i^q + \tilde{c}_i^q - 1 / \tilde{Q}_i^q \right|$$
- $\bar{\tilde{M}}_i$ , calculated as  $\bar{\tilde{M}}_i = \max_{q|q \leq \bar{q}_i, \Delta_i^{\tilde{c}^q} = 1} \{\tilde{M}_i^q\}$ .

### C. DECISION VARIABLES

- $s_i$ , reorder point for item  $i$  [items].
- $Q_i$ , order quantity for item  $i$  [items].
- $\Gamma_i^q$ , one if the  $q$ -th candidate order quantity for item  $i$  is selected, and zero otherwise.
- $s_i^+, s_i^-$ , deviation (down and up, respectively) with respect to initial reorder point for item  $i$  [items].
- $\bar{S}_i^+$ , deviation of item's  $i$  MSL above the average demand during the maximum months of supply for the item [months].
- $f_i, f_i^-$ , fill rate and negative component of fill rate (if any), respectively, for item  $i$  [fraction]. (The negative component is only applicable if  $\tilde{c}_i^q > 1$ .)

$f_{im}^-$ , deficit in fill rate (with respect to target) for item  $i$  in penalty segment  $m$  [fraction].

$z_{in}^{q,SO}$ , ancillary variable for stockouts for item  $i$ , given its  $q$ -th candidate order quantity is chosen and the demand level

$$n \leq n_i^q : z_{in}^{q,SO} = \max \left\{ d_{in}'^q - (s_i - (\tilde{c}_i^q - 1)\tilde{Q}_i^q), 0 \right\}.$$

$\tilde{f}_i, \tilde{f}_i^-$  ancillary binary variables to signal the fill rate sign, if  $\tilde{c}_i^q > 1$ .

#### D. FORMULATION

$$\min_{\mathbf{s}, \mathbf{Q}, \Gamma, \mathbf{s}^+, \mathbf{s}^-, \mathbf{f}, \mathbf{f}^-, \mathbf{z}} \sum_i \sum_m w_{i,m} f_{im}^- + \gamma^p \sum_i \frac{\delta_i^p}{\hat{s}_i^0 + 1.5} (s_i^+ + s_i^-) + \gamma^s \sum_i \frac{\delta_i^s}{\bar{S} + 1} \bar{S}_i^+ \quad (13)$$

subject to:

$$Q_i = \sum_{q|q \leq \bar{q}_i} \tilde{Q}_i^q \Gamma_i^q \quad \forall i \quad (14)$$

$$\sum_{q|q \leq \bar{q}_i} \Gamma_i^q = 1 \quad \forall i \quad (15)$$

$$\tilde{Q}_i^q (1 - (f_i - \Delta_i^{\tilde{c}q} f_i^-)) \leq \sum_{n|n \leq n_i^q} z_{in}^{q,SO} p_{in}'^q + \tilde{Q}_i^q (1 - \Gamma_i^q) \quad \forall i, q | q \leq \bar{q}_i \quad (16)$$

$$\tilde{Q}_i^q (1 - (f_i - \Delta_i^{\tilde{c}q} f_i^-)) \geq \sum_{n|n \leq n_i^q} z_{in}^{q,SO} p_{in}'^q - \bar{d}_i^q (1 - \Gamma_i^q) \quad \forall i, q | q \leq \bar{q}_i \quad (17)$$

$$\tilde{f}_i \geq f_i \quad \forall i | \bar{\Delta}_i^{\tilde{c}} = 1 \quad (18)$$

$$\tilde{f}_i^- \geq f_i^- / \bar{M}_i \quad \forall i | \bar{\Delta}_i^{\tilde{c}} = 1 \quad (19)$$

$$\tilde{f}_i + \tilde{f}_i^- = 1 \quad \forall i | \bar{\Delta}_i^{\tilde{c}} = 1 \quad (20)$$

$$z_{in}^{q,SO} \geq d_{in}'^q - (s_i - (\tilde{c}_i^q - 1)\tilde{Q}_i^q) \quad \forall i, n, q | n \leq n_i^q, q \leq \bar{q}_i \quad (21)$$

$$f_i \geq \bar{f}_i - \sum_m f_{im}^- \quad \forall i \quad (22)$$

$$\sum_i c_i (s_i + Q_i - \lceil \underline{s}_i \rceil \Delta^S) \leq b \quad (23)$$

$$\sum_i \sum_{q|q \leq \bar{q}_i} \frac{\hat{x}_i / \tilde{Q}_i^q}{3t_i} \Gamma_i^q \leq r \quad (24)$$

$$\lceil \underline{s}_i \rceil \Delta^S \leq s_i + Q_i \leq \left\lceil \bar{S}_i^L \frac{\hat{x}_i}{3t_i} \right\rceil \quad \forall i \quad (25)$$

$$\frac{3t_i}{\hat{x}_i} (s_i + Q_i) \leq \bar{S} + \bar{S}_i^+ \quad \forall i \quad (26)$$

$$s_i = \hat{s}_i^0 + s_i^+ - s_i^- \quad \forall i \quad (27)$$

$$\underline{s}_i \leq s_i \leq \bar{s}_i \quad \forall i \quad (\text{optional}) \quad (28)$$

$$Q_i \geq 0 \text{ and integer} \quad \forall i \quad (29)$$

$$s_i \geq -1 \text{ and integer} \quad \forall i \quad (30)$$

$$\Gamma_i^q \in \{0, 1\} \quad \forall i, q \mid q \leq \bar{q}_i \quad (31)$$

$$s_i^+, s_i^-, \bar{S}_i^+ \geq 0 \quad \forall i \quad (32)$$

$$z_{in}^{q, \text{SO}} \geq 0 \quad \forall i, q, n \mid n \leq n_i^q, q \leq \bar{q}_i \quad (33)$$

$$0 \leq f_{im}^- \leq \bar{f}_{im}^- \quad \forall i, m \quad (34)$$

$$\tilde{f}_i, \tilde{f}_i^- \in \{0, 1\} \quad \forall i \mid \bar{\Delta}_i^c = 1 \quad (35)$$

$$f_i, f_i^- \geq 0 \quad \forall i \quad (36)$$

## LIST OF REFERENCES

- GAMS (2016a) General algebraic modeling system. Accessed April 28, 2016, <https://www.gams.com/download/>
- GAMS (2016b) CPLEX 12. Accessed April 28, 2016, <http://www.gams.com/help/index.jsp?topic=%2Fgams.doc%2Fsolvers%2Findex.html>
- Kolodziej S, Grossmann I, Furman K, Sawaya N (2013) A discretization-based approach for the optimization of the multiperiod blend scheduling problem. *Computers & Chemical Engineering* 53: 122-142.
- Naval Supply Systems Command (2016a) Navy supply chain management. Accessed April 17, 2016, <https://www.navsup.navy.mil/navsup/capabilities/nscm>
- Naval Supply Systems Command (2016b) What is NAVSUP? Accessed April 17, 2016, <https://www.navsup.navy.mil/navsup/jobs>
- Naval Supply Systems Command (2016c) NAVSUP weapon systems support (NAVSUP WSS) Accessed May 7, 2016, <https://www.navsup.navy.mil/navsup/ourteam/navsupwss>
- Office of the Chief of Naval Operations (2012) OPNAVINST 4441.12D: Retail supply support of Naval activities and operating forces. Instruction, Office of the Chief of Naval Operations, Washington, DC.
- Roth G (2016) A simulation of alternatives for wholesale inventory replenishment, Master's thesis, Naval Postgraduate School.
- Salmeron J, Craparo E (2015) Wholesale Inventory Optimization Model, Release 2.7. Project Deliverable, Naval Postgraduate School.
- Salmeron J, Craparo E (2016) Site Demand-Based Level Inventory Optimization Model, Release 1.2.3. Project Deliverable, Naval Postgraduate School.
- Silver E, Pkye D, Peterson R (1998) *Inventory Management and Production Planning and Scheduling*. (John Wiley & Sons, New York).
- United States Naval Academy (2016) Commanders Guidance Accessed May 7, 2016, [http://www.usna.edu/Training/\\_files/documents/References/1C%20MQS%20References/NAVSUP%202014%20Commanders%20Guidance.pdf](http://www.usna.edu/Training/_files/documents/References/1C%20MQS%20References/NAVSUP%202014%20Commanders%20Guidance.pdf)

THIS PAGE INTENTIONALLY LEFT BLANK

## **INITIAL DISTRIBUTION LIST**

1. Defense Technical Information Center  
Ft. Belvoir, Virginia
2. Dudley Knox Library  
Naval Postgraduate School  
Monterey, California